Amendment to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (currently amended) A method of audio encoding a stream that carries audio and video data, including:

receiving audio and video data input;

encoding the audio data, for an integer number of *N* audio frames of the audio data, to ~~provide~~ have a mean effective audio frame length $\overline{F}$ that equals a video frame length $1/f_V$ over an ~~integral~~ integer number of M video frames of the video data, where f_V equals a video frame rate of the video data, wherein the encoding includes varying effective audio frame lengths *F* of the audio frames per a respective audio frame index j in a defined sequence of effective audio frame lengths $F(j)$; and

outputting a stream that carries encoded audio and video data, wherein the output stream can be spliced at each video frame without degradation to audio information of the audio data of corresponding audio frames.

2. (currently amended) The method of claim 1, wherein the effective frame length *F* is adjusted by varying an overlap *O* between successive audio frames.

3. (currently amended) The method of claim 1 or claim 2, wherein the value $F(j)$ repeats periodically on *j*, the periodicity of $F(j)$ defining a sequence of audio frames within a sequence of video frames.

4. (previously presented) The method of claim 3 having M video and N audio frames per sequence, each audio frame being composed of *k* blocks of *t* samples each.

5. (currently amended) The method of claim 4, wherein a total overlap O_T between audio frames in the corresponding sequence is equal to $O_T = p \times O + q \times (O+1)$, where O is an overlap length in blocks where $p \in \aleph \wedge q \in \aleph \wedge O \in \aleph \wedge O_T \in \aleph$.

6. (previously presented) The method of claim 5, wherein only audio frames corresponding to a particular video frame are overlapped.

7. (previously presented) The method of claim 6, wherein $p = (N-M) \times (O+1) - O_T$ and $q = (N-M) - p$.

8. (previously presented) The method of claim 5, wherein only audio frames corresponding to a particular video sequence are overlapped.

9. (previously presented) The method of claim 8, wherein $p = (N-1) \times (O+1) - O_T$ and $q = (N-1) - p$.

10. (previously presented) The method of claim 5, wherein any adjacent audio frames are overlapped.

11. (previously presented) The method of claim 10, wherein $p = N \times (O+1) - O_T$ and $q = N - p$.

12. (previously presented) The method of claim 4 in which $\exists n \in \aleph^+ : n \times t = M \times \left(\frac{f_A}{f_V} \right)$.

13. (currently amended) A method of audio encoding a stream that encodes audio and video data, including:

receiving frames of audio and video data;

encoding audio samples of N quasi video-matched audio frames ~~in~~ into frames with a defined sequence of overlap lengths, wherein the encoded audio samples have a mean effective audio frame length $\overline{F}$ that equals a video frame length $1/f_v$ over an integer number of M video frames of the video data, where f_v equals a frame rate of the video data, wherein an effective length of the defined sequence of overlap lengths of the encoded audio frames coincides with a length of a sequence of M video frames, where M and N are positive integers; and

outputting a stream that carries encoded audio and video data, wherein the output stream can be spliced at each video frame without degradation to audio information of the audio data of corresponding audio frames.

14. (previously presented) A data stream encoded by the method of claim 13.

15. (previously presented) The data stream of claim 14, wherein each of the audio frames is tagged to indicate a size of the audio frame.

16. (previously presented) The data stream of claim 14, wherein each block of each audio frame is tagged to indicate whether or not the block is a redundant block.

17. (currently amended) An audio encoder for coding audio for a stream that carries audio and video data, ~~wherein the encoder produces~~ comprising:

an input for receiving audio and video data;

means for encoding the audio data for an integer number of N audio frames into audio frames of variable effective audio frame length such that (i) a mean effective audio frame length $\overline{F}$ of the N audio frames equals (ii) the video frame length $1/f_V$ over an ~~integral~~ integer number of M video ~~and N audio~~ frames, where f_v equals a video frame rate of the video data, and the audio frames of a respective audio frame index j each have a variable overlap that provides an effective audio frame length *F* in a defined sequence of effective audio frame lengths $F(j)$ at encoding; and

an output for outputting a stream that carries the encoded audio and video data, wherein the output stream can be spliced at each video frame without degradation to audio information of the audio data of corresponding audio frames.

18. (currently amended) The audio encoder of claim 17, where the variable overlaps include a total of *p* short overlaps of length *O* and a total of *q* long overlaps of length *O+1* in an overlap sequence, the encoder means further for calculating the overlap sequence using an algorithm that repeats after N frames.

19. (currently amended) An audio decoder for decoding a stream that encodes audio and video data, ~~wherein the decoder calculates~~ comprising:

an input for receiving the stream of encoded audio and video data, wherein the encoded audio data comprises audio data encoded for an integer number of N audio frames into audio frames of variable effective audio frame length such that (i) a mean effective audio frame length $\overline{F}$ of the N audio frames equals (ii) the video frame length $1/f_V$ over an integer number of M video frames, where f_v equals a video frame rate of the video data, and the audio frames of a respective audio frame index j each have a variable overlap that provides an effective audio frame length *F* in a defined sequence of effective audio frame lengths $F(j)$ at encoding;

means for (i) calculating an expected effective audio frame length of an incoming encoded audio frame based on a defined sequence of effective audio frame lengths, ~~adjusts~~ (ii) adjusting the actual length of the incoming encoded audio frame to make it equal to the expected effective audio frame length, ~~determines~~ (iii) determining whether any block within a received encoded audio frame is a redundant block or a non-redundant block, and mapping the non-redundant blocks onto sub-band samples; and

an output for outputting decoded audio and video data obtained in response to a respective processing by the calculating, adjusting, and determining means.

20. (currently amended) The audio decoder of claim 19, ~~wherein the decoder is configured to modify the~~ said means further for (iv) modifying an overlap status of blocks in the data stream by application of one or more of a set of block operators to each block.

21. (previously presented) The audio decoder of claim 20, wherein the set of operators includes one or more of: NOP, an operator that does not change the status of a blocks; DROP, an operator that changes the first non-redundant block from the head overlap into a redundant block; APPEND, an operator that changes the first redundant block from the tail overlap into a non-redundant block; and SHIFT, an operator that is a combination of both DROP and APPEND operators.